3. Gain a comprehensive understanding of vulnerabilities in software and systems.

Comprehensive Understanding of Vulnerabilities in Software and Systems

1. What is a Vulnerability?

A **vulnerability** is a weakness in software, hardware, or system configurations that attackers can exploit to compromise confidentiality, integrity, or availability (CIA Triad).

---

2. Categories of Vulnerabilities

A. Software Vulnerabilities

These originate from flaws in software design, development, or implementation.

Common Software Vulnerabilities:

1. **Buffer Overflow** – Overwriting memory to execute malicious code.
2. **SQL Injection (SQLi)** – Injecting SQL queries to manipulate databases.
3. **Cross-Site Scripting (XSS)** – Injecting malicious scripts into web applications.
4. **Remote Code Execution (RCE)** – Running arbitrary code remotely.
5. **Integer Overflow** – Bypassing limitations by manipulating numerical values.

Causes:

- Poor input validation.
- Lack of secure coding practices.
- Outdated dependencies and libraries.

B. System Misconfiguration Vulnerabilities

These result from insecure settings or improper system setup.

Common System Misconfigurations:

1. **Default Credentials** – Using weak/default passwords (e.g., admin/admin).
2. **Unpatched Software** – Running outdated OS or applications.
3. **Unrestricted Access** – Open ports, public databases, or misconfigured cloud storage.
4. **Weak Permissions** – Overly permissive file and directory access (`chmod 777`).

5. **Excessive Service Exposure** – Running unnecessary services like FTP, Telnet.

Causes:

- Human error in system setup.
- Failure to follow security best practices.
- Lack of monitoring and auditing.

---

3. Vulnerability Scoring & Classification

A. Common Vulnerabilities and Exposures (CVE)

- A standardized list of publicly disclosed vulnerabilities.
- Each vulnerability gets a unique **CVE ID** (e.g., CVE-2021-44228 for Log4Shell).

B. Common Vulnerability Scoring System (CVSS)

- Used to **quantify** vulnerability severity (0-10 scale).
- Factors include:
  - Exploitability (Attack Vector, Complexity, Privileges Required).
  - Impact (Confidentiality, Integrity, Availability).

CVSS Score Breakdown:

Score	Severity Level
0.1 - 3.9	Low
4.0 - 6.9	Medium
7.0 - 8.9	High
9.0 - 10.0	Critical

---

4. How Attackers Exploit Vulnerabilities

A. Exploitation Techniques

1. **Brute Force Attacks** – Cracking weak passwords via automated tools (`hydra`, `john`).
2. **Code Injection** – Injecting malicious code (SQL, XSS, Command Injection).
3. **Privilege Escalation** – Exploiting weak permissions to gain higher access (`sudo -l`, `WinPEAS`).
4. **Man-in-the-Middle (MITM)** – Intercepting and modifying network traffic (`ettercap`, `bettercap`).
5. **Zero-Day Exploits** – Exploiting unknown vulnerabilities before they are patched.

B. Tools Used for Exploitation

Tool	Purpose
`nmap`	Network scanning & service detection
`metasploit`	Exploit framework for RCE & privilege escalation
`sqlmap`	Automates SQL Injection attacks
`Burp Suite`	Web vulnerability testing
`John the Ripper`	Password cracking

---

5. Preventing and Mitigating Vulnerabilities

A. Secure Coding Practices

- ✔ Validate user input (`sanitize & escape`) to prevent injections.
- ✔ Implement proper memory management to avoid buffer overflows.
- ✔ Use secure authentication mechanisms (`bcrypt`, `PBKDF2`).

B. System Hardening

- ✔ Disable unused services and ports.
- ✔ Enforce least privilege access (`sudo`, ACLs).
- ✔ Enable logging and monitoring (`SIEM`, `fail2ban`).

C. Patch Management

- ✔ Regularly update software, firmware, and libraries.
- ✔ Subscribe to vulnerability databases (CVE, NVD).
- ✔ Automate patching via `Ansible` or `WSUS` for Windows.

Conclusion

Understanding vulnerabilities in **software and systems** is key to penetration testing, ethical hacking, and cybersecurity defense. Attackers exploit weaknesses using techniques like **injections, privilege escalation, and network attacks**, but proper **security measures, patching, and system hardening** can significantly reduce risks.